

PARAMETRIC EXCHANGE OF DATA BETWEEN  
A MODELING SYSTEM AND AN EXTERNAL APPLICATION PROGRAM

Technical Field

5           The present invention relates generally to computer systems and more particularly to data exchange between a modeling system and an external application program.

Background of the Invention

10           A computer-aided design (CAD) refers to the use of computers to assist in the design of a product. The designer uses the CAD system to create a model of the product that incorporates critical characteristics of the product. These critical characteristics may include the geometric configuration of the product as well as additional properties and attributes of the product.

15           Most CAD systems have a native data format that is particular to the respective CAD system. Moreover, most CAD systems are largely self-contained in that they only process data residing within the address space of the CAD system. These limitations deter interactive use of external application programs (EAPs) to perform operations relative to the CAD models and other activities on the CAD system.

20

Summary of the Invention

          The present invention addresses the limitations of conventional CAD systems by providing a modeling program/package that supports the exchange of data between an EAP and the modeling program/package. In one embodiment of the present invention,  
25           the modeling program/package is a CAD package that uses parametric feature-based models (which are defined below). An EAP registers with the CAD package to indicate that the EAP receives input data relative to a model in the CAD package and provides output data that is used within the same model or a different model in the CAD package. When there is a need for the EAP to be called, the callback is used to call the EAP as  
30           well as transfer data to the EAP and receive data from the EAP.

          In accordance with one aspect of the present invention, a computer system runs a CAD package and an EAP. A model of an object is provided in the CAD package. The model includes input data to run the EAP and output data from the EAP. The model is

09318105-052499

modified, and it is determined that modifying the model changes the input data to the EAP, requiring recalculation of the output data from the EAP. Thus, new output data is obtained from the EAP. The EAP may, for example, perform an analysis, such as an engineering analysis, on a portion of the model.

5           In accordance with another aspect of the present invention, data is transferred into an EAP from a CAD program. The transferred data is used as the input required by a computation performed by the EAP. Subsequently, the model is modified so as to change the input data to the EAP. The data is automatically transferred by calling the external program without a user request.

10           In accordance with yet another aspect of the present invention, output data is imported into a CAD program from an EAP. The imported output data is integrated into the model. Subsequently, the model is modified so as to require updating of the output data. The output data is automatically updated by calling the EAP without a user request.

15           In accordance with yet another aspect of the present invention, a CAD system includes a CAD program and an EAP that is external to the CAD program. The CAD system also includes a model of an object that contains input data to and output data from the EAP. The CAD system further includes a registration facility for registering the EAP with the CAD program so that the CAD program calls the EAP when the data  
20           in the model to and from the EAP needs updating as a result of changes to the model.

#### Brief Description of the Drawings

An illustrative embodiment of the present invention will be described below relative to the following drawings.

25           FIGURE 1 depicts the exchange of data between an EAP and a CAD package.

FIGURE 2 is a flow chart illustrating the steps that are performed integrating a callback to an EAP and a model in the CAD package

FIGURE 3 is a block diagram illustrating a callback from the CAD package to the EAP to obtain new output data.

30           FIGURE 4A shows an example where the CAD package and the EAP reside within a single computer system.

09318105-052499

FIGURE 4B illustrates an example wherein the CAD package and the EAP reside within separate computer systems.

FIGURE 5 illustrates an exemplary configuration for a computer system for practicing the illustrative embodiment.

5       FIGURE 6 illustrates an example of an assembly that is comprised of multiple parts.

FIGURE 7 illustrates components of a part database in the illustrative embodiment of the present invention.

10       FIGURE 8 is a flow chart illustrating the steps that are performed to modify a model that references an EAP.

FIGURE 9 illustrates an example of the modification of a model that references an EAP

#### Detailed Description of the Invention

15       The illustrative embodiment provides a CAD package that permits data exchange between the CAD package and one or more EAPs. Those skilled in the art will appreciate that the present invention is not limited to implementations where a CAD package is used, but more generally, the present invention may also employ other types of programs that model objects with models that contain geometric and/or numerical  
20       data. Hence, the present invention may also be practiced with computer-aided manufacturing (CAM) programs, CAD/CAM packages, industrial design programs and graphical modeling programs, for example. For purposes of the discussion below, it is presumed that the CAD package is the Pro/ENGINEER 2000i package from Parametric Technology Corporation of Waltham, Massachusetts. The EAP and the CAD package  
25       exchange data parametrically.

      The ability of the illustrative embodiment to integrate EAPs with the CAD package greatly expands the range of functionalities that are available to designers. Moreover, the illustrative embodiment permits designers to customize model definitions by employing custom EAPs. Since the data sent to and received from the EAP is  
30       integrated into the model, the data inside the model that is computed by the EAP is automatically updated when the model is updated, without the user making explicit commands to update the data computed by the EAP.

09318105.052499

The CAD package of the illustrative embodiment uses "feature-based models," which represent a product as a combination of features. A "feature" is a generic shape or characteristic of a product that can be associated with certain attributes and knowledge that is useful for reasoning about the product. Features serve as the building blocks for product definition and for geometric reasoning. Features are modeling entities that allow geometric components of a product to be characterized and associated with a set of attributes that are relevant to an application.

The CAD package employs "parametric models." Parametric models represent models as sets of procedures having input parameters such as dimension values and output geometry. A parametric model stores a procedure for constructing the computer model of a product. In parametric models, the procedure for constructing the product may be viewed as a sequence of assignments to model variables as a function of input parameters for parametric equations. To create parametric variations, the construction procedure is reevaluated after changing the values of the input parameters (such as dimension values). This general definition of "parametric models" includes "variational models" or "variable-driven models" generated by some conventional CAD systems.

For purpose of the discussion below, it is helpful to define a few terms.

A "model" refers to a representation of an object, such as a commercial product or part.

A "geometric object" refers to an entity that has an associated geometry. It is presumed that all of the models described below model geometric objects.

A "part" refers to a component that can be separated from the whole of a product.

An "assembly" refers to a composition including multiple parts.

Figure 1 depicts the basic interaction between an EAP 14 and a CAD package 10 in the illustrative embodiment of the present invention. In particular, the illustrative embodiment permits input data 16 to be sent from the model 12 to the EAP 14 and output data 18 resulting from the computation performed by the EAP 14 to be sent to the CAD package 10. The output data 18 is integrated into a model 12 in the CAD package 10. The output data 18 may take many different forms. For example, the output data may be the results of an engineering analysis. These results might be in the form of numerical parameters and/or geometric entities. Geometric entities include but

09318105-052499

are not limited to points, axes, planes, coordinate systems, curves, surfaces and solids. The model contains features that reference the parameters and/or geometric entities generated by the engineering analysis.

Figure 2 is a flow chart illustrating the steps that are performed in the illustrative embodiment for the EAP 14 to exchange data with the model 12. Initially, the EAP 14 registers with the CAD package 10 (step 20 in Figure 2). Those skilled in the art will appreciate that the registration may take many forms. For example, the EAP 14 may provide information regarding its whereabouts and the appropriate protocol to use in calling the EAP. In the illustrative embodiment, the registration entails putting a callback in the CAD package 10 to call the EAP 14. This callback calls the EAP 14, sends the required input data 16, and obtains the output data 18.

When input data is needed for the EAP 14, the callback for the EAP is called (step 22 in Figure 2). The callback sends the input data 16 required by the EAP. The EAP 14 performs the necessary steps to process the input data 16 and forward the resulting output data 18 to update the model 12 (step 24 in Figure 2). As can be seen in Figure 3, the model 12 contains a callback 30 to the EAP 14. The callback 30 sends new input data 31 to the EAP 14. The EAP 14 generates new output data 32 that is sent to the model 12 and integrated therein.

Those skilled in the art will appreciate that the callback 30 need not call for the entire execution of the EAP 14 but may instead request that a particular function within the EAP be executed. In addition, the callback 30 need not call an application program per se but rather may also call other types of modules instead. Other mechanisms, such as remote procedure calls, may be used instead of callbacks in some embodiments.

Figure 4A depicts a first possible implementation for the illustrative embodiment. In this implementation, a single computer system 50 contains CAD package 10 and an EAP 14. EAP 14 is "external" in that it is outside of CAD package 10 and does not execute within the address space allocated to the CAD package. Figure 4B illustrates an alternative implementation in which the CAD package 10 resides on a first computer system 50A and the EAP 14 resides on a second computer system 50B. These computer systems 50A and 50B may be part of a computer network such as a local area network (LAN) or a wide area network (WAN). For purposes of the

discussion below, it is presumed that the CAD package 10 and EAP 14 reside on a single computer system 50.

Figure 5 is a block diagram illustrating a suitable configuration for the computer system 50. The computer system 50 includes a central processing unit (CPU) that executes computer instructions. The computer system 50 includes a video display 62, keyboard 64, a mouse 66 and an audio output device 68. Those skilled in the art will appreciate that the computer system configuration shown in Figure 5 is intended to be merely illustrative and not limiting of the present invention. The computer system 50 may include different peripheral devices from those shown in Figure 5. Moreover, the computer system 50 may be implemented as a tightly coupled multiprocessor system or even as a distributed system. The computer system 50 may be implemented as a network computer, a personal computer, a mini-computer, a mainframe computer, a super computer, or any of a number of other different types of computer systems.

The computer system 50 includes a network adapter 70 for interfacing with the network 82. The computer system 50 also includes a modem 72 for communicating with remote computing resources over telephone lines, cable lines or wireless communication pathways. The computer system 50 includes a storage 74 that may include both primary memory and secondary memory. The storage 74 may include computer-readable media and removable media such as optical disks, magnetic disks and the like. The storage 74 holds one or more application programs 76. These application programs 76 include the EAP 14. The storage 74 may also hold one or more representations of models 78 for geometric objects. Lastly, the storage 74 may hold a copy of the CAD package 10.

The CAD package 10 represents geometric objects using models that are hierarchical in nature. A model may be an assembly that is comprised of multiple parts. Figure 6 shows an example of an assembly 90 that is composed of parts 92, 94 and 96. Suppose, for example, a designer wishes to generate a model that represents an automobile engine. This automobile engine is an assembly that is comprised of many parts, for example, a four-cylinder engine may include four separate cylinder parts.

Each part within a model contains a number of different types of information. Figure 7 depicts the information that is maintained for a part database 98. The part database includes features 100 as well as a specification of geometry 102. The part

09318105-052499

SLH 4

database holds history information 104, which is maintained to identify history associated with the part. Lastly, the part database contains a procedure to create the part 106. The illustrative embodiment allows one or more features 100 to rely upon output data from the EAP 14. When the feature is being evaluated, the callback is used to call the EAP 14.

Figure 8 is a flow chart illustrating steps that are performed to update a model, where the model has registered a callback to an EAP. Initially, the model is provided (see step 110 in Figure 8). The designer may request a modification to the model (step 112 in Figure 8). Depending on the nature of the modification, the modification may require reevaluation of one or more features, and these features require the calling of the callback (step 114 in Figure 8). The callback sends new input data 16 to the EAP 14, executes the EAP and receives new output data 18 from the EAP 14. The CAD package 10 receives the new output data (step 116 in Figure 8) and updates the model 12 accordingly (step 118 in Figure 8).

Figure 9 illustrates an example in which the EAP 14 performs an analysis and a feature of a model is dependent on the results of the analysis. As shown in Figure 9, the initial model (i.e. stage 130) contains a drill hole that is positioned at the center of gravity of the model. In stage 132, the model changes so as to have a different geometry, resulting in a change in the center of gravity. When the model is being reevaluated to accept change, it is determined that a callback is necessary for the EAP. Hence, in stage 134, the callback is executed and in stage 136, the EAP 14 calculates the new center of gravity. The new center of gravity is returned to the CAD package 10. The drill hole is then shifted to be positioned at the new center of gravity such that in stage 138, the model is fully updated.

While the present invention has been described with reference to an illustrative embodiment thereof, those skilled in the art will appreciate that various changes in form and detail may be made without departing from the intended scope of the present invention as defined in the appended claims.